



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/676,819	09/30/2003	Malte Wedel	09700.0059-00	3069
60668 7590 11/12/2009 SAP / FINNEGAN, HENDERSON LLP 901 NEW YORK AVENUE, NW WASHINGTON, DC 20001-4413				
EXAMINER				
CHEN, QING				
ART UNIT		PAPER NUMBER		
2191				
MAIL DATE		DELIVERY MODE		
11/12/2009		PAPER		

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

# Office Action Summary

**Application No.**

10/676,819

**Applicant(s)**

WEDEL ET AL.

**Examiner**

Qing Chen

**Art Unit**

2191

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 26 June 2009.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-6 and 9-14 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-6 and 9-14 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- 1) ☐ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-8508)  
Paper No(s)/Mail Date \_\_\_\_\_
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: \_\_\_\_\_

**DETAILED ACTION**

1. This Office action is in response to the amendment filed on June 26, 2009.
2. **Claims 1-6 and 9-14** are pending.
3. **Claims 1, 3-6, 9-11, and 14** have been amended.
4. **Claims 7 and 8** have been canceled.
5. The objections to Claims 9-13 are withdrawn in view of Applicant's amendments to the claims.

***Response to Amendment***

***Claim Objections***

6. **Claims 1-14** are objected to because of the following informalities:
  - **Claims 1 and 9** recite the limitation "data processing apparatus." Applicant is advised to change this limitation to read "a data processing apparatus" for the purpose of clarifying the claim language.
  - **Claims 2-6** depend on Claim 1 and, therefore, suffer the same deficiency as Claim 1.
  - **Claims 10-13** depend on Claim 9 and, therefore, suffer the same deficiency as Claim 9.
  - **Claims 1 and 14** recite the limitation "each control." Applicant is advised to change this limitation to read "each control of the plurality of controls" for the purpose of providing it with proper explicit antecedent basis and/or keeping the claim language consistent throughout the claims.

- **Claims 1 and 14** recite the limitations “first user input” and “third user input.”

Applicant is advised to change these limitations to read “a first user input from a user” and “a third user input from the user,” respectively, for the purpose of clarifying the claim language and/or keeping the claim language consistent throughout the claims.

- **Claims 1 and 14** recite the limitation “second input from the user.” Applicant is advised to change this limitation to read “a second user input from the user” for the purpose of clarifying the claim language and/or keeping the claim language consistent throughout the claims.

- **Claims 2-6** depend on Claim 1 and, therefore, suffer the same deficiencies as Claim 1.

- **Claim 4** recites the limitation “a fourth user input.” Applicant is advised to change this limitation to read “a fourth user input from the user” for the purpose of clarifying the claim language and/or keeping the claim language consistent throughout the claims.

- **Claim 9** recites the limitation “user input.” Applicant is advised to change this limitation to read “a user input from a user” for the purpose of clarifying the claim language and/or keeping the claim language consistent throughout the claims.

Appropriate correction is required.

***Claim Rejections - 35 USC § 112***

7. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

8. **Claims 9-14** are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

**Claim 9** recites the limitations “the corresponding data structure” and “the data structure.” There are insufficient antecedent bases for these limitations in the claim. In the interest of compact prosecution, the Examiner subsequently interprets these limitations as reading “the corresponding control data structure” for the purpose of further examination.

**Claims 10-13** depend on Claim 9 and, therefore, suffer the same deficiency as Claim 9.

**Claim 14** recites the limitation “the control.” There is insufficient antecedent basis for this limitation in the claim. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading “each control of the plurality of controls” for the purpose of further examination.

#### ***Claim Rejections - 35 USC § 103***

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

10. **Claims 1, 3, 4, 9, and 11-14** are rejected under 35 U.S.C. 103(a) as being unpatentable over US 5,990,906 (hereinafter “Hudson”) in view of US 6,377,964 (hereinafter “Sano”).

As per **Claim 1**, Hudson discloses:

- displaying a user interface in a client program, the user interface having a plurality of controls, the plurality of controls including multiple types of controls, each control of the plurality of controls having a state and a control data structure, wherein each control data structure corresponds to one control, and wherein the state of each control of the plurality of controls includes a data state and a view state (*see Figure 2; Column 7: 39-45, “As noted above, the undo/redo feature of the present invention is preferably comprised in the LabVIEW graphical programming system from National Instruments Corporation. The LabVIEW graphical programming system includes a large number of different function nodes, structure nodes, and other graphical programming constructs to which the undo/redo feature of the present invention applies.”; Column 9: 52-61, “In step 220 the method initializes data structures for the new transaction that has started. These data structures include the backup list, the type list, and the transaction table.”; Column 12: 33-37, “As shown, for a data change, the method stores the data in the transaction table in step 324. A data change generally involves the data that the user sees, e.g., the value of a number or a front panel control.” and 64-67 to Column 13: 1-4, “As shown, an edit change can comprise either a creation of an object, deletion of an object, or modification of an object. In the preferred embodiment, the method uses a backup list for each transaction for backing up an object in response to an edit change.”; Column 20: 13-15, “In the*

*preferred embodiment, the system allows more than one VI to have undo at a time. Each VI keeps track of its own undo information in its own backup heap.”);*

- for each control in the plurality of controls, storing the state of the control as a first state for the control in the control data structure corresponding to the control (*see Column 10: 41-43, “If an object is required to be modified as determined in step 304, then in step 306 the method stores information regarding the object, i.e., backs up the object.”);*

- receiving a first user input from a user comprising a first change to the state of a first control in the plurality of controls (*see Column 10: 48-51, “In step 308 the graphical programming system applies the change to the object. In other words, in step 308 the user input received in step 202 is applied to perform a transaction or change in the graphical program.”);*

- updating the state of the first control based on the first user input (*see Column 10: 48-51, “In step 308 the graphical programming system applies the change to the object. In other words, in step 308 the user input received in step 202 is applied to perform a transaction or change in the graphical program.”);*

- storing the updated state of the first control as a second state for the first control in the control data structure corresponding to the first control (*see Column 13: 7-11, “The backup list comprises a list of pairs of entries for each transaction, more specifically a list of pairs of ObjIDs. The entries in each pair are referred to as current and previous entries, also referred to as foreground and background entries.”);*

- receiving a second user input from the user comprising a second change to the state of a second control in the plurality of controls (*see Column 10: 48-51, “In step 308 the graphical*

*programming system applies the change to the object. In other words, in step 308 the user input received in step 202 is applied to perform a transaction or change in the graphical program.”);*

- updating the state of the second control based on the second user input (*see Column 10: 48-51, “In step 308 the graphical programming system applies the change to the object. In other words, in step 308 the user input received in step 202 is applied to perform a transaction or change in the graphical program.”);*

- storing the updated state of the second control as a second state for the second control in the control data structure corresponding to the second control (*see Column 13: 7-11, “The backup list comprises a list of pairs of entries for each transaction, more specifically a list of pairs of ObjIDs. The entries in each pair are referred to as current and previous entries, also referred to as foreground and background entries.”);*

- receiving a third user input from the user comprising a request to undo the first change to the first control (*see Column 15: 45-48, “As shown, in step 402 the user selects the undo or redo option.”);*

- determining whether the first change affects the data state of the first control (*see Column 16: 1, “In step 410 the method undoes data changes.”);*

- determining whether the first change affects the view state of the first control (*see Column 15: 54, “In step 408, the method undoes edit changes.”);*

- restoring the state of the first control to reflect the first state for the first control without affecting the state of the second control (*see Column 15: 1-8, “Therefore, in the preferred embodiment, when a fixed sized object is being backed up, the newly created object is the one that goes into the background, and the original object stays in the foreground and is*



*modified. Since the original object remains at the same memory location, no referencing objects are required to be modified. Accordingly, no other object is required to change its reference to the object, and no further backups are required.” and 54, “In step 408, the method undoes edit changes.”; Column 16: 1, “In step 410 the method undoes data changes.”); and*

- clearing the stored first state for the first control and the stored second state for the first control from the control data structure corresponding to the first control without affecting the control data structure corresponding to the second control (*see Column 19: 43-48, “For example, if the user changes the type of a control on a connector pane, or changes the connector pane pattern, then all of the callers are automatically updated to use that new calling information. As part of that process of automatically updating, all of the callers are directed to throw away their undo information.”*).

However, Hudson does not disclose:

- transmitting the restored state of the first control to a server.

Sano discloses:

- transmitting a restored state of a first control to a server (*see Column 10: 55-60, “It is assumed designer A executes the undo function in this state. This operation corresponds to the cancel command for the operation of “deletion of element 2” in this example according to FIG. 30. This operation is interpreted by the undo function execution unit 14 and transmitted to the database server 20 via the network 1.”*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Sano into the teaching of Hudson to modify Hudson’s invention to include transmitting the restored state of the first control to a server. The

modification would be obvious because one of ordinary skill in the art would be motivated to store the history of user operations in a separate database in order to limit the amount of memory/storage used in the user's computer system (*see Sano – Column 5: 14-45*).

As per **Claim 3**, the rejection of **Claim 1** is incorporated; and Hudson further discloses:

- restoring the state of the first control only if the first change affects the data state of the first control (*see Column 16: 1, "In step 410 the method undoes data changes."*).

As per **Claim 4**, the rejection of **Claim 1** is incorporated; and Hudson further discloses:

- receiving a fourth user input from the user comprising a request to redo the first change to the first control (*see Column 15: 45-48, "As shown, in step 402 the user selects the undo or redo option."*); and

- restoring the state of the first control to reflect the second state for the first control (*see Column 15: 54, "In step 408, the method undoes edit changes."; Column 16: 1, "In step 410 the method undoes data changes."*).

As per **Claim 9**, Hudson discloses:

- generating a plurality of data structures that store application data and associations between the application data and a plurality of application controls, wherein each application control of the plurality of application controls has a state and a control data structure, wherein each control data structure corresponds to one application control of the plurality of application controls, wherein the state of each application control of the plurality of application controls

includes a data state and a view state, and wherein each application control of the plurality of application controls is rendered based on the application data (see Figure 2; Column 7: 39-45, “As noted above, the undo/redo feature of the present invention is preferably comprised in the LabVIEW graphical programming system from National Instruments Corporation. The LabVIEW graphical programming system includes a large number of different function nodes, structure nodes, and other graphical programming constructs to which the undo/redo feature of the present invention applies.”; Column 9: 52-61, “In step 220 the method initializes data structures for the new transaction that has started. These data structures include the backup list, the type list, and the transaction table.”; Column 12: 33-37, “As shown, for a data change, the method stores the data in the transaction table in step 324. A data change generally involves the data that the user sees, e.g., the value of a number or a front panel control.” and 64-67 to Column 13: 1-4, “As shown, an edit change can comprise either a creation of an object, deletion of an object, or modification of an object. In the preferred embodiment, the method uses a backup list for each transaction for backing up an object in response to an edit change.”; Column 20: 13-15, “In the preferred embodiment, the system allows more than one VI to have undo at a time. Each VI keeps track of its own undo information in its own backup heap.”);

- detecting that a first application control of the plurality of application controls has changed from a first prior state to a first new state (see Column 10: 48-51, “In step 308 the graphical programming system applies the change to the object. In other words, in step 308 the user input received in step 202 is applied to perform a transaction or change in the graphical program.”);

- determining whether the change affects the data state of the first application control  
(see Column 16: 1, “In step 410 the method undoes data changes.”);
- determining whether the change affects the view state of the first application control  
(see Column 15: 54, “In step 408, the method undoes edit changes.”);
- recording, for the first application control, the first prior state of the first application control in the corresponding control data structure (see Column 10: 41-43, “If an object is required to be modified as determined in step 304, then in step 306 the method stores information regarding the object, i.e., backs up the object.”);
- updating, for the first application control, the corresponding control data structure of the plurality of data structures based on the first new state (see Column 13: 7-11, “The backup list comprises a list of pairs of entries for each transaction, more specifically a list of pairs of ObjIDs. The entries in each pair are referred to as current and previous entries, also referred to as foreground and background entries.”);
- detecting that a second application control of the plurality of application controls has changed from a second prior state to a second new state (see Column 10: 48-51, “In step 308 the graphical programming system applies the change to the object. In other words, in step 308 the user input received in step 202 is applied to perform a transaction or change in the graphical program.”);
- recording, for the second application control, the second prior state of the second application control in the corresponding control data structure (see Column 10: 41-43, “If an object is required to be modified as determined in step 304, then in step 306 the method stores information regarding the object, i.e., backs up the object.”);

- updating, for the second application control, the corresponding control data structure of the plurality of data structures based on the second new state (*see Column 13: 7-11, "The backup list comprises a list of pairs of entries for each transaction, more specifically a list of pairs of ObjIDs. The entries in each pair are referred to as current and previous entries, also referred to as foreground and background entries."*);

- receiving a user input from a user requesting that an undo operation be performed on the first application control (*see Column 15: 45-48, "As shown, in step 402 the user selects the undo or redo option."*);

- performing the undo operation by restoring the first application control to the first prior state without affecting the state of the second application control (*see Column 15: 1-8, "Therefore, in the preferred embodiment, when a fixed sized object is being backed up, the newly created object is the one that goes into the background, and the original object stays in the foreground and is modified. Since the original object remains at the same memory location, no referencing objects are required to be modified. Accordingly, no other object is required to change its reference to the object, and no further backups are required."* and 54, *"In step 408, the method undoes edit changes."*; Column 16: 1, *"In step 410 the method undoes data changes."*);

- updating, for the first application control, the corresponding control data structure of the plurality of data structures based on the first prior state (*see Column 13: 7-11, "The backup list comprises a list of pairs of entries for each transaction, more specifically a list of pairs of ObjIDs. The entries in each pair are referred to as current and previous entries, also referred to as foreground and background entries."*); and

- clearing, for the first application control, the stored application data in the corresponding control data structure of the plurality of data structures without affecting the control data structure corresponding to the second application control (*see Column 19: 43-48, "For example, if the user changes the type of a control on a connector pane, or changes the connector pane pattern, then all of the callers are automatically updated to use that new calling information. As part of that process of automatically updating, all of the callers are directed to throw away their undo information."*).

However, Hudson does not disclose:

- transmitting the restored first prior state of the first application control to a server.

Sano discloses:

- transmitting a restored first prior state of a first application control to a server (*see Column 10: 55-60, "It is assumed designer A executes the undo function in this state. This operation corresponds to the cancel command for the operation of "deletion of element 2" in this example according to FIG. 30. This operation is interpreted by the undo function execution unit 14 and transmitted to the database server 20 via the network 1."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Sano into the teaching of Hudson to modify Hudson's invention to include transmitting the restored first prior state of the first application control to a server. The modification would be obvious because one of ordinary skill in the art would be motivated to store the history of user operations in a separate database in order to limit the amount of memory/storage used in the user's computer system (*see Sano – Column 5: 14-45*).

As per **Claim 11**, the rejection of **Claim 9** is incorporated; and Hudson further discloses:

- wherein at least one data structure of the plurality of data structures is stored on a client device (*see Figure 1: 12*).

As per **Claim 12**, the rejection of **Claim 9** is incorporated; and Hudson further discloses:

- wherein the plurality of application controls include multiple types of controls (*see Column 7: 39-45, "As noted above, the undo/redo feature of the present invention is preferably comprised in the LabVIEW graphical programming system from National Instruments Corporation. The LabVIEW graphical programming system includes a large number of different function nodes, structure nodes, and other graphical programming constructs to which the undo/redo feature of the present invention applies."*).

As per **Claim 13**, the rejection of **Claim 9** is incorporated; and Hudson further discloses:

- wherein the associations between the application data and the plurality of application controls are defined by metadata (*see Column 9: 52-61, "In step 220 the method initializes data structures for the new transaction that has started. These data structures include the backup list, the type list, and the transaction table."*).

**Claim 14** is an apparatus claim corresponding to the computer program product claim above (Claim 1) and, therefore, is rejected for the same reason set forth in the rejection of Claim 1.

11. **Claim 2** is rejected under 35 U.S.C. 103(a) as being unpatentable over **Hudson** in view of **Sano** as applied to Claim 1 above, and further in view of “**HTML 4.01 Specification**,” **December 1999** (hereinafter “**HTML1999**”).

As per **Claim 2**, the rejection of **Claim 1** is incorporated; however, Hudson and Sano do not disclose:

- wherein the multiple types of controls include one or more of a text field control type, a radio button control type, a table control type, a tray control type, and a menu control type.

HTML1999 discloses:

- wherein multiple types of controls include one or more of a text field control type, a radio button control type, a table control type, a tray control type, and a menu control type (*see Section 17.2.1*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of HTML1999 into the teaching of Hudson to modify Hudson's invention to include wherein the multiple types of controls include one or more of a text field control type, a radio button control type, a table control type, a tray control type, and a menu control type. The modification would be obvious because one of ordinary skill in the art would be motivated to enhance usability of an online web form.



12. **Claims 5 and 6** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Hudson** in view of **Sano** as applied to Claim 1 above, and further in view of **US 6,167,455 (hereinafter "Friedman")**.

As per **Claim 5**, the rejection of **Claim 1** is incorporated; however, Hudson and Sano do not disclose:

- wherein the third user input comprising the request to undo the first change is received while focus is not on the first control.

Friedman discloses:

- wherein a third user input comprising a request to undo a first change is received while focus is not on a first control (*see Column 2: 36-44, "The individual command objects are linked, so that one command object can be accessed and invoked in one context, when the other command object is invoked in an active context. This allows for synchronization of the execution of the command objects, so that both command objects are either done or undone at the same time. In this manner, the user will perceive the action as unified, even though it affects data objects in two contexts."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Friedman into the teaching of Hudson to modify Hudson's invention to include wherein the third user input comprising the request to undo the first change is received while focus is not on the first control. The modification would be obvious because one of ordinary skill in the art would be motivated to produce consistent results when a user attempts to undo/redo different objects (*see Friedman – Column 2: 64-67*).

As per **Claim 6**, the rejection of **Claim 1** is incorporated; however, Hudson and Sano do not disclose:

- wherein restoring the state of the first control includes restoring a state of another control that shares data with the first control.

Friedman discloses:

- wherein restoring the state of a first control includes restoring a state of another control that shares data with the first control (*see Column 2: 44-47, "The user can thus cause the do and undo method of one command object to be invoked, and the corresponding do or undo method of a linked command object will also be invoked."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Friedman into the teaching of Hudson to modify Hudson's invention to include wherein restoring the state of the first control includes restoring a state of another control that shares data with the first control. The modification would be obvious because one of ordinary skill in the art would be motivated to produce consistent results when a user attempts to undo/redo different objects that share the same data (*see Friedman – Column 2: 64-67*).

13. **Claim 10** is rejected under 35 U.S.C. 103(a) as being unpatentable over **Hudson** in view of **Sano** as applied to Claim 9 above, and further in view of US **6,543,006** (hereinafter "**Zundel**").

As per **Claim 10**, the rejection of **Claim 9** is incorporated; however, Hudson and Sano do not disclose:

- wherein at least one of the plurality of data structures is at least one data tree.

Zundel discloses:

- wherein at least one of a plurality of data structures is at least one data tree (*see Column 4: 40-46, "Program 30 utilizes several Directed Acyclic Graph (DAG) data structures to track design data and design intent. These structures will be briefly discussed to provide a foundation for terminology used throughout this description. These DAGs are presented purely for exemplary purposes--other data structures, such as non-directional graphs, trees, etc., can also be used."*; *Column 8: 60 and 61, "Note also that either a linear list or hierarchical tree can be used to track operations and related program states."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Zundel into the teaching of Hudson to modify Hudson's invention to include wherein at least one of the plurality of data structures is at least one data tree. The modification would be obvious because one of ordinary skill in the art would be motivated to utilize a tree structure to make data easier to manipulate and search.

#### ***Response to Arguments***

14. Applicant's arguments filed on June 26, 2009 have been fully considered, but they are not persuasive.

***In the Remarks, Applicant argues:***

a) Hudson in view of Sano does not teach or suggest at least these elements of claim 1. In particular, Hudson teaches an undo/redo method for graphical programming which minimizes data storage. See Hudson, Abstract. The method uses a backup list for each transaction for backing up an edit change, which comprises a list of paired "ObjIDs" which indicate the current and previous entries. See Hudson, 13:5-9. The method also uses a backup list list, that is, a list containing all the generated backup lists. See Hudson, 13:11-13. When undoing an action, Hudson teaches undoing the last action performed, that is, undoing the last change made, which affects the last backup list in the backup list list. See Hudson 15:43-16:6. Hudson's system cannot restore the state of a first control without affecting the state of a second control that was changed at a later time because the second control action is a later action, and thus comes last on Hudson's list. Accordingly, Hudson does not teach the above-quoted claim 1 recitations.

***Examiner's response:***

a) Examiner disagrees. With respect to the Applicant's assertion that Hudson's system cannot restore the state of a first control without affecting the state of a second control that was changed at a later time, as previously pointed out in the Non-Final Rejection (mailed on 04/21/2009) and further clarified hereinafter, the Examiner respectfully submits that Hudson clearly discloses "restoring the state of the first control to reflect the first state for the first control without affecting the state of the second control" (*see Column 15: 1-8, "Therefore, in the preferred embodiment, when a fixed sized object is being backed up, the newly created object is the one that goes into the background, and the original object stays in the foreground and is modified. Since the original object remains at the same memory location, no referencing objects*

*are required to be modified. Accordingly, no other object is required to change its reference to the object, and no further backups are required.” and 54, “In step 408, the method undoes edit changes.”; Column 16: 1, “In step 410 the method undoes data changes.”).* Note that Hudson’s invention is directed to an undo/redo feature for a graphical programming system which includes a hierarchy of interconnected objects. When an object (*e.g.*, fixed size) is being modified, a copy of the object is backed up in the background and the object remains in the foreground for modifying. Thus, any other objects that reference the object do not need to change their reference to the object and are unaffected by the restoration of the object’s original state.

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 1, 9, and 14 are proper and therefore, maintained.

***In the Remarks, Applicant argues:***

b) Additionally, claim 1 is also allowable for another reason. The Office Action alleged that Hudson teaches “clearing the stored first state for the control and the stored second state for the control from the control data structure,” as recited in claim 1. See Office Action, p. 15. But the Office Action misinterpreted the scope and content of the Hudson reference. Hudson discloses an undo limit that allows the user to set a maximum number of undos, and purges any undo information beyond this limit. See Hudson, 9:24-30. If, for example, the user set the undo limit to “5” then undo “6” and “7” may be purged. This does not, however, teach “clearing the stored first state for the first control and the stored second state for the first control from the control data structure corresponding to the first control.” In order to clear the stored first state for the first control and the stored second state for the first control, Hudson’s undo limit would have to be set

to zero, which would disable the undo functionality. In fact, Hudson expressly states that "[i]f the minimum number of steps is 0, undo is disabled". Hudson, 8:11-12. Disabling the undo function teaches away from the claimed invention, and further, would render the Hudson system unfit for its intended purpose. Moreover, Sano does not cure this deficiency. Accordingly, for this additional reason, the Office Action does not establish a prima facie case of obviousness for claim 1 based on the combination of Hudson and Sano.

***Examiner's response:***

b) Examiner disagrees. With respect to the Applicant's assertion that Hudson does not teach "clearing the stored first state for the first control and the stored second state for the first control from the control data structure corresponding to the first control without affecting the control data structure corresponding to the second control," the Examiner respectfully submits that Hudson clearly discloses "clearing the stored first state for the first control and the stored second state for the first control from the control data structure corresponding to the first control without affecting the control data structure corresponding to the second control" (*see Column 19: 43-48, "For example, if the user changes the type of a control on a connector pane, or changes the connector pane pattern, then all of the callers are automatically updated to use that new calling information. As part of that process of automatically updating, all of the callers are directed to throw away their undo information."*).

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 1, 9, and 14 are proper and therefore, maintained.

***Conclusion***

15. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

16. Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Qing Chen whose telephone number is 571-270-1071. The Examiner can normally be reached on Monday through Thursday from 7:30 AM to 4:00 PM. The Examiner can also be reached on alternate Fridays.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Wei Zhen, can be reached on 571-272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Q. C./

Examiner, Art Unit 2191

/Wei Y Zhen/

Supervisory Patent Examiner, Art Unit 2191